CSCI 410 Pattern Recognition

Assignment One

by Denton Bobeldyk

Part I Single Feature

1. Create a matlab script that will perform each of the steps required for this exercise.
2. Load ‘partOneData.mat’ into the matlab environment (included in blackboard as part of the assignment).
3. Create a histogram for each of the class distributions {classOne, classTwo}. Plot each of the histograms on the same figure (use 100 bins). The figure should contain a title, legend and the x and y axis should be labeled appropriately.
4. Report the prior probability of classOne? (Hint: Number of classOne samples divided by all samples)
5. Report the prior probability of classTwo? (Hint: see above hint, but for classTwo)
6. Create a random partition of the data, splitting each of the classes into 60% training and 40% testing.
    a. Using only the training data, find the maximum likelihood estimator for the following parameters:
        i. $Class\ One: \mu, \sigma$
        ii. $Class\ Two: \mu, \sigma$
    b. Classify each of the samples in the test partition using a Bayesian classifier (you must create a function that will do this). Report the prediction accuracy for class one and class two.

Hint: You will need to create a method that, given the mean and standard deviation of a distribution, determines the probability that ‘x’ belongs to the distribution.

Matlab template below:

```
function probability = computeGaussianDensity(mean, stdDev, x)

≪Your code to calculate the Gaussian density here ≫


end
```

Part II Multivariate

1. Create a matlab script that will perform each of the steps required for this exercise.
2. Load ‘partTwoData.mat’ into the matlab environment (included in blackboard as part of the assignment).
3. Report the prior probability of classOne?
4. Report the prior probability of classTwo?
5. Create a random partition of the data, splitting each of the classes into 60% training and 40% testing.
    a. Using only the training data, find the maximum likelihood estimator for the following parameters:
        i. $Class\ One: \mu, covariance\ matrix$
        ii. $Class\ Two: \mu, covariance\ matrix$
    b. Classify each of the test samples using a Bayesian classifier (you must create a function that will do this). Report the prediction accuracy for each class.

Hint: You will need to create a method that, given the mean and covariance matrix, determines the probability the vector ‘x’ belongs to the distribution.

Matlab template below:

```
function probability = computeGaussianDensityMultivariate(mean, covarianceMatrix, x)

≪Your code here≫


end
```

Turn-in:

1. A single Matlab script that completes part one and part two
    a. You should use ‘fprintf’ statements to output the answers to each of the above questions.
2. A Word document that contains a screenshot of the histogram generated in part one step three as well as a copy and paste of the output your matlab script generated.